Orchestrating Code Deployments to Put the Service Into SaaS

Customer Site
Chicago, IL
United States

About the Company

Company:	Fieldglass, Inc.
Industry:	Software
Installation:	Chicago, Illinois, United States
ASCI Product:	ActiveBatch® Workload Automation & Job Scheduling

Brief Company Overview:
Fieldglass is committed to providing the market leading Software as a Service (SaaS) platform to help companies successfully procure and manage contingent workers, services such as Statement of Work projects, independent contractors, and specialized talent pools.

Used by Global 2000 companies such as GlaxoSmithKline, Johnson & Johnson, Monsanto and salesforce.com, our customers span industries including financial services, healthcare, pharmaceuticals, high technology and telecommunications.

FIELDGLASS™

Case Study Highlights

✓ Working in a 24/7 world
✓ Providing Non-Production Deployment Services and Visibility With Reduced Learning Curve
✓ Non-Technical Users Can See What Types of Jobs They Can Run In Certain Types of Non-Production Environments
✓ Remove Tedious Housekeeping Scripts and Implement a Job Library
✓ Quality of Advanced Systems Concepts Technical Support

A Lot of Data and Code Developments

It used to be that companies could measure their workforce by how many heads they counted in their offices or factories. No longer. Today, human capital is a complex mix of job sharing, home-based employees, contract/temporary workers, outsourcing firms, consultants and more, in addition to traditional full-time hires.

To plan and manage this fragmented new world of work, many companies retain Fieldglass®, a technology solution leader for the acquisition of contingent workers, service providers and direct hires. Its Web-based, Software as a Service (SaaS) application is considered to be the first to coordinate the entire spectrum of human capital acquisition.

The Fieldglass solution is continually updated and tested for operational quality. In addition, Fieldglass conducts customized tests for individual customers on a regular basis. David Hegland, senior configuration manager for Fieldglass, notes that this ambitious commitment dictates the need for dozens of continuously updated non-production development environments. "We have a 24/7 streak a mile wide here. Things have to work, all the time," he says. "With four major releases and several service packs per year, we're constantly creating and changing our nonproduction environments for testing, demonstrations, and pilot projects. And that means a lot of data and code deployments."

The non-production environments are leveraged for everything from quality assurance and testing to sales demos. At one time, Fieldglass had just a few non-production environments and senior staff members used a combination of scripts, the operating system's native scheduled task feature, and manual intervention to execute routine deployment tasks. "Everybody would pile into a conference room and we'd manually update these environments," Hegland says.

ActiveBatch Becomes the Middleman

Over the years, the number of environments has grown alongside the company. "Beyond scheduling, our challenge has been to provide scalable non-production deployment services and visibility without teaching everyone the intricacies of deployment," Hegland says.

"But ActiveBatch serves as the on-demand triggering mechanism to a very large, less technical audience. It puts the trigger in their hands and allows them to monitor jobs in real time without getting involved in the difficulties associated with the actual code deployment."

To rectify the problem, Fieldglass implemented ActiveBatch in 2003 to serve initially as a scheduler, but later as the "middleman" between the company's in-house code deployment solution and the Fieldglass team. "We have an in-house code deployment solution that does the actual deployment, which is a set of activities that are technical, tricky and fraught with difficulty," Hegland says. "But ActiveBatch serves as the on-demand triggering mechanism to a very large, less technical audience. It puts the trigger in their hands and allows them to monitor jobs in real time without getting involved in the difficulties associated with the actual code deployment."

"Non-technical users can instantly see what types of jobs they can run in certain types of environments, trigger those jobs and monitor their progress, all through a simple web page. It's really powerful stuff."

ActiveBatch is Simple and Makes Processes Simple

A large part of this is due to the simplicity of ActiveBatch, Hegland says. As a result, Fieldglass has more than 50 users who leverage ActiveBatch in one capacity or another. "We have a wide variety of different types of deployments-related operations in the non-production environments that we like to run with our in-house deployment solution. So I've set up a folder within ActiveBatch with a template for each of those job types. Then I've got another set of folders, one for each of the 80 non-production environments, and references to the templates from those folders. A simple web page uses the ActiveBatch API to display all of the available actions for an environment and includes links to trigger those actions. "Non-technical users can instantly see what types of jobs they can run in certain types of environments, trigger those jobs and monitor their progress, all through a simple web page. It's really powerful stuff."

As Fieldglass' use of ActiveBatch has progressed through the years, the company has witnessed the maturity of the product as a mission-critical part of an IT environment. Hegland commented how the large and continuously growing jobs library has allowed his staff to package jobs quickly for nontechnical users without the "hidden cost" of writing bullet-proof scripts themselves. "ActiveBatch's job library is very orderly and controlled," observed Hegland. "We make extensive use of variables and reference jobs. ActiveBatch's library eliminates virtually all of the tedious housekeeping scripts we would have to write—and once the workflow is set, it's done the same way every time."

"ActiveBatch's job library is very orderly and controlled. We make extensive use of variables and reference jobs. ActiveBatch's library eliminates virtually all of the tedious housekeeping scripts we would have to write—and once the workflow is set, it's done the same way every time."

Hegland's team executes daily code builds and hundreds of deployments. ActiveBatch enables the team to package jobs that support both operations. Restoring and backing up production data, subject to stringent requirements for its handling, is a much simpler process as well. "ActiveBatch enables us to coordinate and multi-thread those actions," Hegland noted.

Hegland also pointed out the quality of technical support he receives for ActiveBatch. "Advanced Systems Concepts' tech support is one of the best support teams I've ever worked with, and that's saying a lot. They know exactly what steps to take, any time we call. As Fieldglass is constantly striving to provide the best possible service to our customers, we are pleased to work with a vendor who clearly thinks likewise."

"Advanced Systems Concepts' tech support is one of the best support teams I've ever worked with, and that's saying a lot. They know exactly what steps to take, any time we call. As Fieldglass is constantly striving to provide the best possible service to our customers, we are pleased to work with a vendor who clearly thinks likewise."